Joint Engineering White Paper

Using Oracle Database 10g's Automatic Storage Management with EMC Storage Technology

Nitin Vengurlekar, Oracle Corporation
Bob Goldsand, EMC Corporation

Updated 5/3/2005

ORACLE

EMC²
where information lives

Part Number H1144

Table of Contents

Purpose

The new Oracle Database 10g Automatic Storage Management (ASM) feature is focused on simplifying the management of the storage used for holding database information. The database information includes the database files, as well as other information such as database configuration information, backup and log archives, etc.

Prior to Oracle 10g and the introduction of ASM, it has typically been the joint function of the database administrator, the system administrator, and the storage administrator to plan physical storage layout to be used for housing the various pieces of information making up the mission critical databases. With ASM, the task is simplified in that the database administrators can now focus on acquiring and provisioning database storage, and relegating the actual data placement mapping responsibility to the Oracle database kernel.

EMC and Oracle have conducted comprehensive engineering work to date and published best practice storage layout and general database file allocation guidelines that are widely adopted by many of our joint customers. System administrators and DBAs who are familiar and comfortable working with the current data planning and placement strategies, will need to see the benefits and advantages that ASM brings to their database environment prior to converting existing database environments to an ASM managed pool of storage. Once they are familiar with the concepts and understand the advantages of this environment they will want to see validation that it works well with a joint EMC and Oracle technology stack.

In order to harness the full advantage of ASM together with EMC storage, EMC and Oracle engineering have worked jointly to refine some of the details of storage configuration and layout best practice guidelines to cover the new ASM deployment paradigm. This paper shares the best practice recommendations based on the results of our joint engineering work.

Scope

The intention of this document is to outline best practices and the factors that will affect the performance and manageability of the Oracle Database 10g ASM feature in an EMC Symmetrix/DMX array.

Introduction

When laying out a database, administrators must consider many storage configuration options. The storage solution must facilitate high performance I/O. It must provide protection against failure of storage hardware, such as disks and host bus adaptors. Growing and changing workloads require a dynamic storage configuration. The automation of storage related tasks, reduces the risk of human error.

This paper explains how to combine Oracle's Automatic Storage Management with EMC storage technology to meet these challenges. It first describes the relevant products, features, and terminology from EMC and Oracle. It then outlines techniques to leverage ASM and EMC to meet the challenges of modern database storage administration.

EMC Technologies

The following section provides an overview and introduction to the various techniques and technologies used to configure a Symmetrix storage array.

Metavolumes

A metavolume is a single host-addressable volume that is comprised of concatenated or striped hypervolumes. The following outlines some of the benefits of metavolumes:

- Distribute I/O across back-end disks.
- Reduce the number of devices to manage
- Make it possible to use a large numbers of logical disks with fewer LUNs per port

Striping/Double Striping

Striping is a technique available with metavolumes that can provide significant performance benefits by spreading I/O load across multiple disk spindles. With a striped metavolume, the addresses are interleaved across all members of the metavolume at the granularity specified when the metavolume was created. Striped metavolumes can reduce I/O contention on the physical disks, since even requests with a high locality of reference will be distributed across the members of the metavolume.

A technique called double striping, also known as plaiding or stripe-on-stripe can be implemented by host-based striping of striped-metavolumes.

Several performance tests have been performed in the past indicate that double striping works well for most all workload characteristics.

EMC PowerPath

Although ASM does not provide multi-pathing capabilities, ASM does leverage multi-pathing tools. EMC PowerPath provides this multi-pathing capability.

EMC PowerPath is host-resident software for storage systems to deliver intelligent I/O path management and volume management capabilities. With PowerPath, administrators can improve the server's ability to manage batch processing or a changing workload through continuous and intelligent I/O channel balancing. PowerPath automatically configures multiple paths and dynamically tunes performance as workloads change.

EMC PowerPath offers the following features and benefits:
- **Multipath support** — Provides multiple channel access.
- **Load balancing** — Automatically adjusts data routing for optimum performance and eliminates the need to statically configure devices among multiple channels.
- **Path failure** — Automatically and non-disruptively redirects data to an alternate data path; eliminates application downtime in the event of path failure.
- **Online recovery** — Allows you to resume use of a path after the path is repaired without service interruption.

It is highly recommended to implement PowerPath along with ASM.

Oracle Technologies

The following section outlines the Oracle technology that is addressed in this white paper. As the focus of this paper is specifically addressing one feature of the new Oracle 10g database, the only Oracle technologies addressed here are Automatic Storage Management and Flash Recovery Area.

Automatic Storage Management

In Oracle10^g, storage management and provisioning for the database has becomes much more simplified with a new feature called *Automatic Storage Management* (*ASM)*. ASM provides filesystem and volume manager capabilities built into the Oracle database kernel. With this capability, ASM simplifies storage management tasks, such as creating/laying out databases and diskspace management.

Designed specifically to simplify the job of the database administrator (DBA), ASM provides a flexible storage solution that simplifies the management of a dynamic database environment. The features provided by ASM make most manual I/O performance tuning tasks unnecessary. ASM automates best practices and helps increase the productivity of the DBA.

Since ASM allows disk management to be done using familiar create/alter/drop SQL statements, DBAs do not need to learn a new skillset or make crucial decisions on provisioning. Additionally, ASM operations can be completely managed with 10^g Enterprise Manager.

To use ASM for database storage, you must create one or more ASM disk groups, if none exist already. A disk group is a set of disk devices that ASM manages as a single unit. ASM spreads data evenly across all of the devices in the disk group to optimize performance and utilization.

In addition to the performance and reliability benefits that ASM provides, it can also increase database availability. You can add or remove disk devices from disk groups without shutting down the database. ASM automatically rebalances the files across the disk group after disks have been added or removed. Disk groups are managed by a special Oracle instance, called an ASM instance. This instance must be running before you can start a database instance that uses ASM for storage. When you choose ASM as the storage mechanism for your database, DBCA creates and starts this instance if necessary.

Flash Recovery Area

The Flash Recovery Area is a unified storage location for all recovery related files and activities in an Oracle database. By defining one init.ora parameter, all RMAN backups, archive logs, control file autobackups, and datafile copies are automatically written to a specified file system or ASM disk group. In addition, RMAN automatically manages the files in the Flash Recovery Area by deleting obsolete backups and archive logs that are no longer required for recovery. Allocating sufficient space to the Flash Recovery Area will ensure faster, simpler, and automatic recovery of the Oracle database.

The Flash Recovery Area provides:

- Unified storage location of related recovery files
- Management of the disk space allocated for recovery files
- Simplified database administration tasks
- Much faster backup
- Much faster restore
- Much more reliability

Configuring Automatic Storage Management

This section describes the steps necessary to configure an ASM instance and disk groups using external redundancy.

ASM Instances

In Oracle Database 10g there are two types of instances: database and ASM instances. The database instance will be discussed in a later section. The ASM instance, which is generally named +ASM, is started with the INSTANCE_TYPE=ASM init.ora parameter. This parameter, when set, signals the Oracle initialization routine to start an ASM instance and not a standard database instance. Unlike the standard database instance, the ASM instance contains no physical files; such as logfiles, controlfiles or datafiles, and only requires a few init.ora parameters for startup.

Upon startup, an ASM instance will spawn all the basic background processes, plus some new ones that are specific to the operation of ASM. STARTUP clauses for ASM instances are similar to those for database instances. For example, RESTRICT prevents database instances from connecting to this ASM instance. NOMOUNT starts up an ASM instance without mounting any disk group. MOUNT option simply mounts all defined diskgroups[1].

The illustration in Figure 1 shows an initialized ASM instance. Observe that all ASM processes begin with *asm,* as opposed to the database instance, whose processes begin with *ora*.

[1] The OPEN startup option performs the same function as MOUNT option; i.e., mount all asm_diskgroups.

ASM is the file and storage manager for all databases [that employ ASM] on a given node. Therefore, only one ASM instance is required per node regardless of the number of database instances on the node. Additionally, ASM seamlessly works with the RAC architecture to support clustered storage environments. In RAC environments, there will be one ASM instance per clustered node, and the ASM instances communicate with each other on a peer-to-peer basis.

ASM init.ora Parameters

```
*.instance_type=asm
*.large_pool_size=12M
*.asm_diskstring='/dev/raw/raw*'
```

ASM SGA and parameter sizing

Enabling the ASM instance is as simple as configuring a handful of init.ora parameters. The init.ora parameters specified in Figure 1 are the essential parameters required to start up ASM.

- db_cache_size - This value determines the size of the cache. This buffer cache area is used to cache metadata blocks. The default value suit most all implementations.
- shared_pool - Used for standard memory usage (control structures, etc.) to manage the instance. Also used to store extent maps. The default value suit most all implementations.
- large_pool - Used for large allocations. The default values suit most all implementations.

The processes init.ora parameter for ASM may need to be modified. The recommendations pertain to versions 10.1.0.3 and later of Oracle, and will work for RAC and non-RAC systems. The following formula can used to determine an optimal value for this parameter:

25 + 15n, where n is the number of databases using ASM for their storage.

Access to the ASM instance is comparable to a standard instance; i.e., SYSDBA and SYSOPER. Note however, since there is no data dictionary, authentication is done from an Operating System level and/or an Oracle password file. Typically, the SYSDBA privilege is granted through the use of an operating system group. On Unix, this is typically the dba group. By default, members of the dba group have SYSDBA privileges on all instances on the node, including the ASM instance. Users who connect to the ASM instance with the SYSDBA privilege have complete administrative access to all disk groups in the system. The SYSOPER privilege is supported in ASM instances and limits the set of allowable SQL commands to the minimum required for basic operation of an already-configured system. The following commands are available to SYSOPER users:

- STARTUP/SHUTDOWN
- ALTER DISKGROUP MOUNT/DISMOUNT
- ALTER DISKGROUP ONLINE/OFFLINE DISK
- ALTER DISKGROUP REBALANCE
- ALTER DISKGROUP CHECK
- Access to all V$ASM_* views

All other commands, such as CREATE DISKGROUP, ADD/DROP/RESIZE DISK, and so on, require the SYSDBA privilege and are not allowed with the SYSOPER privilege.

Instance Name

Instance type

```
SQL> select instance_name from v$instance

INSTANCE_NAME
----------------
+ASM
```

ASM Background Processes

Figure 1.

```
oracle    2423     1  0 Apr30 ?        00:00:00 asm_pmon_+ASM1
oracle    2425     1  0 Apr30 ?        00:00:00 asm_diag_+ASM1
oracle    2427     1  0 Apr30 ?        00:00:01 asm_lmon_+ASM1
oracle    2430     1  0 Apr30 ?        00:00:36 asm_lmd0_+ASM1
oracle    2432     1  0 Apr30 ?        00:00:01 asm_lms0_+ASM1
oracle    2434     1  0 Apr30 ?        00:00:00 asm_mman_+ASM1
oracle    2436     1  0 Apr30 ?        00:00:00 asm_dbw0_+ASM1
oracle    2438     1  0 Apr30 ?        00:00:00 asm_lgwr_+ASM1
oracle    2440     1  0 Apr30 ?        00:00:00 asm_ckpt_+ASM1
oracle    2442     1  0 Apr30 ?        00:00:00 asm_smon_+ASM1
oracle    2444     1  0 Apr30 ?        00:16:03 asm_rbal_+ASM1
oracle    2447     1  0 Apr30 ?        00:00:17 asm_lck0_+ASM1
oracle    2457     1  0 Apr30 ?        00:00:00 oracle+ASM1
(DESCRIPTION=(LOCAL=
```

Cluster Synchronization Services

ASM was designed to work with single instance as well as with RAC clusters. ASM, even in single-instance, requires that Cluster Synchronization Services (CSS) is installed and available. In a single instance CSS maintains synchronization between the ASM and database instances. CSS, which is a component of Oracle's Cluster Ready Services (CRS), is automatically installed on every node that runs Oracle Database 10^g. However, in RAC environments, the full Oracle Cluster-ware (CRS) is installed on every RAC node.

Since CSS provides cluster management and node monitor management, it inherently monitors ASM and its shared storage components (disks and diskgroups). Upon startup, ASM will register itself and all diskgroups it has mounted, with CSS. This allows CSS across all RAC nodes to keep diskgroup metadata in-sync. Any new diskgroups that are created are also dynamically registered and broadcasted to other nodes in the cluster.

As with the database, internode communication is used to synchronize activities in ASM instances. CSS is used to heartbeat the health of the ASM instances. ASM internode messages are initiated by structural changes that require synchronization; e.g. adding a disk. Thus, ASM uses the same integrated lock management infrastructure that is used by the database for efficient synchronization.
Database instances contact the CSS to lookup the TNS connect string (using the diskgroup name as an index). This connect string is then used to create a persistent connection into the ASM instance.

Disks Groups

A disk group is a collection of disks managed as a logical unit. ASM spreads each file evenly across all disks in the disk group to balance the I/O. A disk group is comparable to a LVM's volume group or a storage group. The ASM file system layer, which transparently sits atop the disk group, is not visible to O/S users. The ASM files are only visible to the Oracle database kernel and related utilities. Disk group creation is best done with close coordination of the EMC storage administrator and the System Administrator. The storage administrator will identify a set of disks from the storage array. Each OS will have its unique representation of disk naming.

Disks

The first task in building the ASM infrastructure is to discover and associate (adding) disks under ASM management. This step is best done with some coordination of the Storage and Systems administrators. The Storage administrator will identify a set of disks from the storage array that will be presented to the host. The term *disk* may be used in loose terms. A disk can be partition of a physical spindle or refer to the entire spindle itself, this depends on how the storage array presents the logical unit number (LUN) to the Operating System (OS). In this document we will refer generically to LUNs or disks presented to the OS as simply, *disks.* On Linux systems, disks will generally have the following SCSI name format: */dev/sdxy*.

In SAN environments, it assumed that the disks are appropriately identified and configured; i.e., they are properly zoned and LUN masked within the SAN fabric and can be seen by the OS. Once the disks are identified, they will need to be discovered by ASM. This requires that the disk devices (Unix filenames) have their ownership changed from root to oracle. These candidate disks must already have a valid label on it (if necessary), and should not be currently managed (encapsulated) by any other logical volume manager (LVM) such as Veritas. Having a valid label on the disk prevents inadvertent or accidental use of the disk.

Disk Discovery

Once disks are identified, they must be discovered by ASM. This requires that the oracle user have read/write permission for the disk devices (OS filenames).
When ASM scans for disks, it will use the asm_diskstring parameter to find any devices that it has permissions to open. Upon successful discovery, the V$ASM_DISK view will now reflect which disks were discovered. Notice, that the name is empty and the group_number is set to 0. Disks that are discovered, but not yet associated with a diskgroup have a null name, and a group number of 0.

In our example on Linux (syntax may vary by platform), disks raw1, raw2, and raw3 were identified, and their ownership changed to oracle. These disks can be defined in the init.ora parameter ASM_DISKSTRING. In our example we used the following setting for ASM_DISKSTRING:

```
'/dev/raw/raw*'
```

This invokes ASM to scan all the disks that match that string, and find any Oracle owned devices. The V$ASM_DISK view will now show which disks were identified.

```
SQL> select name, path, header_status, state, disk_number from
v$asm_disk

NAME         PATH                    HEADER_ST  STATE    DISK_NUMBER
------------ ---------------------- ---------- -------- -----------
             /dev/raw/raw1           CANDIDATE  NORMAL             0
             /dev/raw/raw2           CANDIDATE  NORMAL             1
             /dev/raw/raw3           CANDIDATE  NORMAL             2
```

Creating Disk Groups

The creation of a diskgroup involves the validation of the disks to be added. These disks cannot already be in use by another diskgroup, must not have a pre-existing ASM header, and cannot have an Oracle file header. This prevents ASM from destroying an existing data device. Disks with a valid header status, which include candidate, former, or provisioned, are only ones allowed to be diskgroups members

Once the disks are identified and readied, a disk group can be created that will encapsulate one or more of these drives.

In order to create disk group DG_ASM, we assume disk discovery has returned the following devices:

```
/dev/raw/raw1
/dev/raw/raw2
/dev/raw/raw3
```

The following statement will create disk group DG_ASM using external redundancy with the above disk members. External redundancy disk groups rely on RAID protection from the storage hardware.

```
create diskgroup DG_ASM external redundancy disk
‘/dev/raw/raw1’ NAME ata_disk1,
‘/dev/raw/raw2’ NAME ata_disk2,
‘/dev/raw/raw3’ NAME ata_disk3;
```

The NAME clause is optional when creating disk groups. If a name is not specified, then a default name will be assigned. It is advisable to use a descriptive name when creating disk groups; doing so will simplify management of many disk groups. This also makes it easier to identifying disks when altering existing disk groups.

Adding Disks to a Disk Group

The ADD clause of the ALTER DISKGROUP statement enables you to add disks to a diskgroup

```
ALTER DISKGROUP DG_ASM ADD DISK
'/dev/raw/raw4' NAME ata_disk4;
```

Dropping Disks from Disk Groups

To drop disks from a disk group, use the `DROP DISK` clause of the `ALTER DISKGROUP` statement.

```
ALTER DISKGROUP DG_ASM DROP DISK ata_disk4;
```

Note that the `ALTER DISKGROUP DROP` statement references the disk name and not the discovery device string. When performing `an ALTER DISKGROUP ADD` or `DROP`, ASM will automatically rebalance the files in the disk group.

Rebalance and Redistribution

With traditional volume managers, expansion or reduction of the striped file systems has typically been difficult. Using ASM, these disk/volume activities have become seamless and redistribution (rebalancing) of the striped data can now be performed while the database remains online.

Any storage configuration changes will trigger a rebalance. The main objective of the rebalance operation is to distribute each data file evenly across all storage in a disk group. Because of ASM’s extent placement algorithm, ASM does not need to re-stripe all of the data during a rebalance. To evenly redistribute the files and maintain a balanced I/O load across the disks in a disk group, ASM only needs to move an amount of data proportional to the amount of storage added or removed.

Note, since rebalancing involves physical movement of file extents, this introduces some level of impact to user-online community. To minimize this impact a new init.ora parameter has been introduced. The init.ora parameter, ASM_POWER_LIMIT (applied only to ASM instance), provides rebalance throttling, so the impact to the online access of the database instance can be managed. There is a trade-off between speed of redistribution and impact. A value of 0 is valid and will stop rebalance. A value of 11 is full throttle with more impact, whereas a value of 1 is low speed and low impact.

Monitoring Rebalance Process

```
SQL"> alter diskgroup DG_ASM add disk '/dev/raw/raw5' rebalance power 11;


SQL"> select * from v$asm_operation

OPERA  STAT      POWER     ACTUAL      SOFAR   EST_WORK   EST_RATE EST_MINUTES

-----  ---- ---------- ---------- ---------- ---------- ---------- -----------

1 REBAL WAIT         11          0          0          0          0           0

1 DSCV  WAIT         11          0          0          0          0           0

(time passes……..)

OPERA STAT      POWER     ACTUAL      SOFAR   EST_WORK   EST_RATE EST_MINUTES

------------ ----- ---- ---------- ---------- ---------- ---------- ----------

1 REBAL REAP         11         11         25        219        485           0
```

.

ASM Recovery

Since ASM manages the physical access to ASM files and its metadata, a shutdown of the ASM instance will cause the client database instances to shutdown as well

In a single ASM instance configuration, if the ASM instance fails while disk groups are open for update, after the ASM instance restarts, it recovers transient changes when it mounts the disk groups. In RAC environments, with multiple ASM instances sharing disk groups, if one ASM instance should fail (that is a RAC node fails), another node's ASM instance automatically recovers transient ASM metadata changes caused by the failed instance.

Joint Best Practices for ASM with EMC Storage

In addition to the standard best practices for running Oracle Databases on EMC storage technologies, there are some additional joint best practices that are recommended for the use of Automatic Storage Management on EMC technology.

Use External Redundancy Disk Groups for EMC LUNs

ASM external redundancy disk groups were designed to leverage RAID protection provided by storage arrays, such as the Symmetrix. EMC supports several types of protection against loss of media and provides transparent failover in the event of a specific disk or component failure. Offloading the overhead task of providing redundancy protection will increase the CPU cycles of the database server that will improve its performance. As such, it is best to use EMC LUNs in ASM disk groups configured with external redundancy.

Create Disk Groups with Similar Capacity and Performance LUNs

When building ASM disk groups, or extending existing ones, it is best to use LUNs (striped metavolumes, hyper volumes or logical disks) of similar size and performance characteristics. As with striped metavolumes, the performance of the group will be determined by its slowest member.

When managing disks with different size and performance capabilities, best practice is to group them into disk groups according to their characteristics. ASM distributes files across the available storage pool based on capacity, and not the number of spindles. For example, if a disk group has three LUNs, two of 50GB and a one of 100GB, the files will be distributed with half of their extents on the bigger LUN and the remaining half split evenly between the two smaller drives. This would yield sub-optimal performance, because the larger disk would perform twice as many I/Os. When each LUN in a disk group is of a similar size, ASM spreads files such that each LUN performs an equal number of I/Os.

For most database environments that use the Oracle 10g Flash Recovery Area, it might be common to find two ASM disk groups: one comprised of the fastest storage for the database work area while the Flash Recovery Area disk group might have lower performance storage (like the inner hyper volumes or ATA drives).

Use More Than One LUN per ASM Disk Group

As ASM distributes the Oracle database files across the pool of storage in each disk group, it is best practice to enable this level of distribution to work across more than just a single LUN. As striped metavolumes provide an even distribution of I/O across the back end of the storage layer and PowerPath provides a dynamic distribution across the available channels, use of ASM to evenly distribute the database reads and writes across members of the storage pool can further decrease the chances of an I/O bottleneck. ASM providing the distribution of I/O at the server side while EMC metavolumes provides the striping of the I/O on the storage side is referred to as double striping.

Testing of this combination has shown that these technologies are complementary and provide further performance enhancement as well as ease of manageability for both the DBA and SA. The optimal number of LUNs per disk group depends on how many host bus adaptors are available. Four to eight LUNs per disk group is a good rule of thumb.

Note: When configuring ASM for large environments for use with split mirror technology (TimeFinder or SRDF), the creation of several disk groups should be considered. As ASM requires an atomic split of all disks in the disk group, having more than one enables administrators to manage smaller granularity with more flexibility to replicate only parts of that environment.

Place Log Files and Datafiles in the same Diskgroup

In a normal database operation mode, the redo logs are fairly write-intensive. As such, the redo logs have traditionally been placed on their own physical disk drives. When striped metavolumes are used in conjunction with ASM across several LUNs, the activity to these hot devices is distributed in a manner that leverages the full extent of the server and storage infrastructure resource. Therefore redo logs and datafiles may reside on the same diskgroup when using ASM and striped metavolumes.

Use PowerPath with ASM for Improved HA and Performance

PowerPath provides multipath support by presenting one pseudo-device per LUN (and one native device per physical path) to the host operating system. In order to minimize administrative errors, it is highly recommended that the PowerPath pseudo-device names be used and not the native path-names for ASM disk groups. Examples of native names are /dev/sda, etc.

Additionally, it is a best practice to always do a fdisk and create a partition that starts one megabyte into the LUN presented to the OS. Having a partition provides a method to track a device using fdisk, and prevents accidental use of a disk. The one megabyte offset is to preserve alignment, between ASM stripping and storage array internal stripping.

The following is an example of EMC PowerPath pseudo devices used in raw device administrative tasks in a Linux environment:

- Use the PowerPath `powermt display dev=all` command to determine pseudo names. In our example we want to use pseudo name emcpowerg.

```
$powermt display dev=all

Pseudo name=emcpowerg
Symmetrix ID=000187430301
Logical device ID=0011
state=alive; policy=SymmOpt; priority=0; queued-IOs=0
==============================================================================
--------------- Host ---------------   - Stor -   -- I/O Path -  -- Stats -
### HW Path                I/O Paths    Interf.   Mode    State  Q-IOs
Errors
==============================================================================
1  QLogic Fibre Channel 2300 sdbx       FA 4aA   active  alive       0  0
2  QLogic Fibre Channel 2300 sdcf       FA 4aB   active  alive       0  0
```

- Using fdisk, create a partition on that device that is 1Mb offset into the disk.

- Create the raw device using PowerPath pseudo name.
  - `raw /dev/raw/raw10 /dev/emcpowerg1`

- To maintain these raw device bindings upon reboot, they must be entered into the `/etc/sysconfig/rawdevices` file as follows:
  - `/dev/raw/raw10 /dev/emcpowerg1`

- To guarantee that the raw device binding occurs during any restart, use the `chkconfig` utility.
  - `/sbin/chkconfig rawdevices on`

Since the host sees a striped metavolume as a single LUN, a low queue-depth could result in metavolume members not having enough I/Os queued against them. With PowerPath installed, you have multiple active paths to the meta device, which effectively increases the number of commands queued to the volume; making queue depth problem unlikely.

10g RAC clusters and Powerpath.

Although there are no specific configuration requirements for 10gRAC and PowerPath or Symmetrix, there are some items that need to be addressed. There may be cases where the path names (PowerPath, or native devices) which are not consistent across nodes of a cluster. For example, a Symmetrix disk device may be known to node1 as emcpowera, whereas, on node2 the same Symm device will addressed as emcpowerj. This is not an issue for ASM, since ASM does not require that the disks have the same names on every node. However, ASM does require that the same disk be visible to each ASM instance via that instances' discovery string. The instances can have different discovery strings if required. Use the PowerPath command powermt display dev=all, to find the common Logical Device (logical ids). The Two paths [from two different nodes] will reference the same physical Symm device if they have the same PowerPath logical id.
This is illustrated below, emcpowerg on node1 and emcpowera on node2 will reference the same Symm device since they share the same logical device id, *Logical device ID=015D.*

```
#### on Node1
$powermt display dev=all

Pseudo name=emcpowerg
Symmetrix ID=000187430301
Logical device ID=0009
state=alive; policy=SymmOpt; priority=0; queued-IOs=0
==============================================================================
--------------- Host ---------------   - Stor -   -- I/O Path -  -- Stats -
--
### HW Path                I/O Paths    Interf.   Mode    State  Q-IOs
Errors
==============================================================================
1  QLogic Fibre Channel 2300 sdax       FA 13cA   active  alive      0  0
2  QLogic Fibre Channel 2300 sddf       FA 14cA   active  alive      0  0


####Node 2
Pseudo name=emcpowerg
Symmetrix ID=000187430301
Logical device ID=0011
state=alive; policy=SymmOpt; priority=0; queued-IOs=0
==============================================================================
--------------- Host ---------------   - Stor -   -- I/O Path -  -- Stats -
### HW Path                I/O Paths    Interf.   Mode    State  Q-IOs
Errors
==============================================================================
1  QLogic Fibre Channel 2300 sdbx       FA 4aA   active  alive       0  0
2  QLogic Fibre Channel 2300 sdcf       FA 4aB   active  alive       0  0
```

There are cases when consistent path names across RAC clustered nodes is desired[2]. The following PowerPath configuration files are keys to this procedure, as they contain the path naming associations.

- PowerPath 3.0.2, emcpower.conf file, located in /etc/opt/emcpower.
- PowerPath 4.3, /etc/emcp_deviceDB.dat and /etc/emcp_deviceDB.idx files.

To implement consistent PowerPath path names, install PowerPath on all nodes of the cluster. After the installation, shutdown PowerPath on all but one node. Then, the copy the correct PowerPath configuration file(s), based on the appropriate PowerPath version, to all the other nodes, making sure the orginal files are renamed or removed. Then, upon reboot or restart of PowerPath service, the new PowerPath configuration files will generate the appropriate name assignment, assuming the same set of storage objects are in fact discovered on the other nodes.

[2] Currently this procedure works on Linux and Solaris systems. This procedure will not work on AIX and Windows systems, due to the manner in which disks are registered to the OS.

Conclusion

The introduction of Automatic Storage Management in the Oracle database 10g greatly reduces the administrative tasks associated with managing Oracle database files. ASM is a fully integrated host level file system and volume manager for Oracle and eliminates the need for third party volume management for database files. ASM enables

- Automation of the best practice file naming conventions
- Dynamic disk configuration while the database remains online
- Complementary technology to the EMC storage technologies

Striped metavolumes spread the I/O evenly across their members, resulting in improved performance. EMC RAID provides protection against disk failure. ASM external redundancy disk groups with Metavolumes protected by EMC RAID provide reliable, performance-tuned storage for Oracle databases.

PowerPath provides seamless multipath failover capabilities and is complementary to ASM for both high availability and performance.

ASM provides a simplified storage management interface for the DBA while EMC technologies provide ease of storage provisioning and storage management for the System Administrator. These technologies can be combined in several ways to greatly reduce the cost of managing the entire environment. The combined technology enables simplified provisioning and management of the storage resources for the database. This makes DBAs and SAs more productive by consolidating touch points, reducing the overlap of responsibilities, and removing some dependencies.

Following these best practices for ASM with EMC storage technologies will provide significant savings in management costs from the database, systems, and storage administration perspective.

Since 1995, EMC and Oracle have invested to jointly engineer and integrate their technologies. The combination of Oracle and EMC software and best practices used in an integrated fashion can greatly reduce the cost of designing, implementing and operating your IT infrastructure.

Appendix 1: Using EMC TimeFinder with ASM

EMC TimeFinder allows customers to use business continuance volume (BCV) devices. A BCV is a full copy of a Symmetrix standard device. BCVs can be accessed by one or more separate hosts while the standard devices are online for normal I/O operations from their host(s). A standard device is established with a BCV, making a BCV pair. The BCV pair can be split, reestablished (resynchronized), and restored. When BCVs are split from the standard devices, the BCVs represent a mirror of the standard device as of that point in time, and can be accessed directly by a host.

EMC TimeFinder can create point in time copies of an ASM disk group. This technique and methodology is consistent with non ASM Oracle databases. However, when creating split mirror snapshots of ASM diskgroups, TimeFinder Consistent Split technology must be employed to provide an atomic split of all disks (LUNs) in the disk group.
Additionally, the database must be put into hot backup mode prior to the split of the BCV volumes. These split BCVs, can then be mounted on the backup or secondary host. This backup or secondary host must have ASM installed and configured, so that ASM can discover the disks and mount the diskgroup. Note, it is not permissible to mount BCVs on the same host where the standard volumes are mounted.
RMAN must be used to create backup of the BCV copy of the database.

It is a best practice to minimize the ASM activity through the duration of this procedure. This ASM activity includes adding or dropping disks, or manually invoking a rebalance operation.

The following procedure will illustrate the method from creating and establishing the EMC device group to appropriately splitting the EMC device group.

Configuring EMC TimeFinder

The following steps should be performed prior to running any Oracle database commands:

1. Determine the device status
2. Create the devices
3. Associate the devices

The following section describes each step.

Determine Device Status

If you are creating BCV pairing relationships from devices that have never been paired, the process is relatively straightforward. However, if previous pairings may exist from old device groups, check the status of devices before adding them to your group

To check whether standards or BCVs already belong to device groups, use the `symdev list` command:

```
symdev list
```

To check whether standards or BCVs already have pairing relationships, use the `symbcv list` command:

```
symbcv list
```

Create a Device Group for the Data and Flash Area ASM diskgroups.

Create a device group named ProdDataASM for the Data Diskgroup:

```
symdg create ProdASM –type regular
```

Create a device group named ProdFlashASM for the Flash Diskgroup:

```
symdg create ProdFlashASM –type regular
```

ASM manages all the datafiles associated with a given database in a disk group. In order to create the device group we must determine which disks are contained in each disk group. Every disk must be identified, or the disk group copy will be incomplete.

The following query is used to determine the necessary associations. In our example it is known that ASM group number '1' is Data and '2' is the Flash.

```
SQL> select path from v$asm_disk where group_number=1;

PATH
---------------------
/dev/raw/raw1
/dev/raw/raw2
/dev/raw/raw3

SQL> select path from v$asm_disk where group_number=2;

PATH
---------------------
/dev/raw/raw4
/dev/raw/raw5
/dev/raw/raw6
```

Once the disks have been identified you can add standard devices /dev/raw/raw1-3 to the device group named ProdASM.

```
symld -g ProdDataASM  add pd /dev/raw/raw1 DEV01
symld -g ProdDataASM  add pd /dev/raw/raw2 DEV02
symld -g ProdDataASM  add pd /dev/raw/raw3 DEV03
```

Add standard devices /dev/raw/raw4-6 to the device group named ProdFlashASM.

```
symld -g ProdFlashASM  add pd /dev/raw/raw4 DEV04
symld -g ProdFlashASM  add pd /dev/raw/raw5 DEV05
symld -g ProdFlashASM  add pd /dev/raw/raw6 DEV06
```

Associate BCV's

Associate BCVs with the ProdDataASM and ProdFlashASM device groups. A default logical name will be assigned to the BCVs:

```
# For Data diskgroup
symbcv -g ProdDataASM  associate DEV088
symbcv -g ProdDataASM  associate DEV089
symbcv -g ProdDataASM  associate DEV08A

# For Flash diskgroup
symbcv -g ProdFlashASM  associate DEV08B
symbcv -g ProdFlashASM  associate DEV08C
symbcv -g ProdFlashASM  associate DEV08D
```

Configure the Target Host

Prior to establishing the BCVs, the source database init.ora file must be copied to the target host database directory (`ORACLE_HOME/dbs/initdb1.ora`).

In addition an ASM instance must be configured and running on the target host. In the init+ASM.ora file (or spfile+ASM.ora) the BCV devices must be discoverable via the ASM_DISKSTRING parameter described earlier. The target host may have a different device name than the source host. To determine the appropriate ASM_DISKSTRING entry, run the following command:

```
syminq | egrep '88| 89|8A'  or  sympd list

Symmetrix ID: 000184600051

   Device Name              Directors            Device
--------------------------- ------------ ------------------------


Physical                      Sym  SA :P DA :IT  Config
--------------------------- ------------ ----------------------
/dev/raw/raw11               0088   14A:0 01A:C2  BCV
/dev/raw/raw12               0089   14A:0 01B:C1  BCV
/dev/raw/raw13               008A   14A:0 02B:D0  BCV
```

Re-run the above command for the Flash BCVs.

Note*:* `syminq` and `sympd list` commands will return the entire physical disk device designator, be sure you use the proper partition in the discovery string.

Establishing a BCV

Establish TimeFinder BCVs prior to placing the database in hot backup mode. Use the `–full` option (full copy) the first time devices are paired. Use incremental establish (copy only changed tracks) otherwise. Verify that TimeFinder is fully synchronized before the next step.

```
symmir -g ProdDataASM -full establish

symmir -g ProdFlashASM -full establish
```

Placing Database in Hot Backup Mode

Before performing a BCV split for backup the database must be placed in hot backup mode. In order to place the database in hot backup mode, the database must have archive logging enabled.

- First flush the current redo information by issuing a log switch:

```
SQL>alter system archive log current;
```

- To place database in Hot Backup Mode:

```
SQL> alter database begin backup;
```

Performing a TimeFinder Split in Oracle:

With the database in backup mode split the BCV volumes containing the Oracle ASM disk groups. Note, a consistent split is the only split supported for ASM disk group splits.

```
symmir –g ProdDataASM split –consistent –noprompt
```

After completing the TimeFinder split the database can be taken out of hot backup mode.

```
SQL> alter database end backup;
```

Flush the current redo information by issuing a log switch:

```
SQL>alter system archive log current;
```

```
symmir –g ProdFlashASM split –consistent –noprompt
```

The database can now be made available on the target host as a point-in-time copy. As mentioned previously ASM must be configured on the target host. When ASM is started up it will automatically mount the disk group. If ASM is already running perform the following command:

```
SQL>alter diskgroup DATA mount;

SQL>alter diskgroup FLASH mount;
```

```
Startup the database and perform recovery
```

Appendix 2: Related Documents

- *EMC: Oracle Database Layout on Symmetrix Enterprise Storage Systems*
- *Oracle: Oracle Database 10g Automatic Storage Management Technical Best Practices, Nitin Vengurlekar*
- *EMC/Oracle: Joint best practices performance and tuning EMC/Oracle*
- *EMC: Using SYMCLI to Perform TimeFinder Control Operations* (P/N 300-000-074)